Representing Syntax

By Roger Keays, 29 February 2008

I have been searching for a concise and accurate way to record the syntax of a language. One that you might be able to feed into a computer such that it only produces and parses grammatical sentences. However, during my search, I came across something in particular which really makes me doubt the analytical approach to defining languages.

Most ideas of representing syntax attempt to define a grammar which generates a tree where each node restricts the class of phrases or words which can appear beneath it. E.g. NP = (Det) (AP) N (PP).

This works pretty well, except for some tricky substitutions such as *Ed is on the JPA expert group, but Iamon the JSF one*. Here, one only replaces part of the noun phrase *"the JPA expert group"*. To handle these cases, linguists invented X-bar theory.

X-bar theory basically inserts nodes into the tree so that you always end up with a binary tree. E.g:

```
NP = (Det) N'
N' = AP N'
N' = N (PP)
```

It solves the problem, because one can now replace an N'. However, these feels just *too* academically convenient. With a binary tree structure like this, it is possible to replace *any* sequence of consecutive tokens. It makes me think that, while a tree is a useful construct, it isn't reflecting what is really going on with the language. Did our brain really evolve into a binary tree?!

Other things that a tree grammar alone cannot indicate are relations (such as subject an object), transitivity, agreement and thematic role requirements. Some of this information could be included as additional rules separate from the grammar, but it hardly seems like a silver bullet.

Another major concern I have with tree based grammars is the identification of word boundaries. What is so special about words that makes us separate morphology from syntax? Do words really exist at all? In writing we have whitespace, but in speaking there are no pauses between words. Words can be identified because they are easily substitutable, but we've seen that with syntax trees too.

So what now? Honestly, I have no idea. Keep reading I guess. I found one interesting article which included some ideas of the fractal nature of language [1]. Something along the lines that each word has its own grammar which is best defined by pattern in the brain which it is represented by.

So, if language is simply a pattern in the brain, should we recognize lexemes, morphemes, syntax, semantics and pragmatics at all? Are these inherent parts of the pattern of language, or has language been cleaved into these boxes by linguists? (murderers!)

I used to think all languages were the same because so many of them are so similar, but now I'm starting to think that this is only because we've been doing the same things with language for so long. As a sort of counter-example, consider the Pirahs who are reportedly unable to learn to count because their language doesn't have the necessary features to make sense of numbers (such as recursion) [2]. So what are we unable to do because of the limitations of *our* language?

A lot, I think.

References

[1] http://informatics.indiana.edu/rocha/univgram.html

[2] http://en.wikipedia.org/wiki/Pirah_language

About Roger Keays

Roger Keays is an artist, an engineer, and a student of life. He has no fixed addressand has leftfootprints on 40-something different countries around the world. Roger is addicted to surfing. His other interests are music, psychology, languages, the proper use of semicolons, and finding good food.

« Extraspection

Back to English